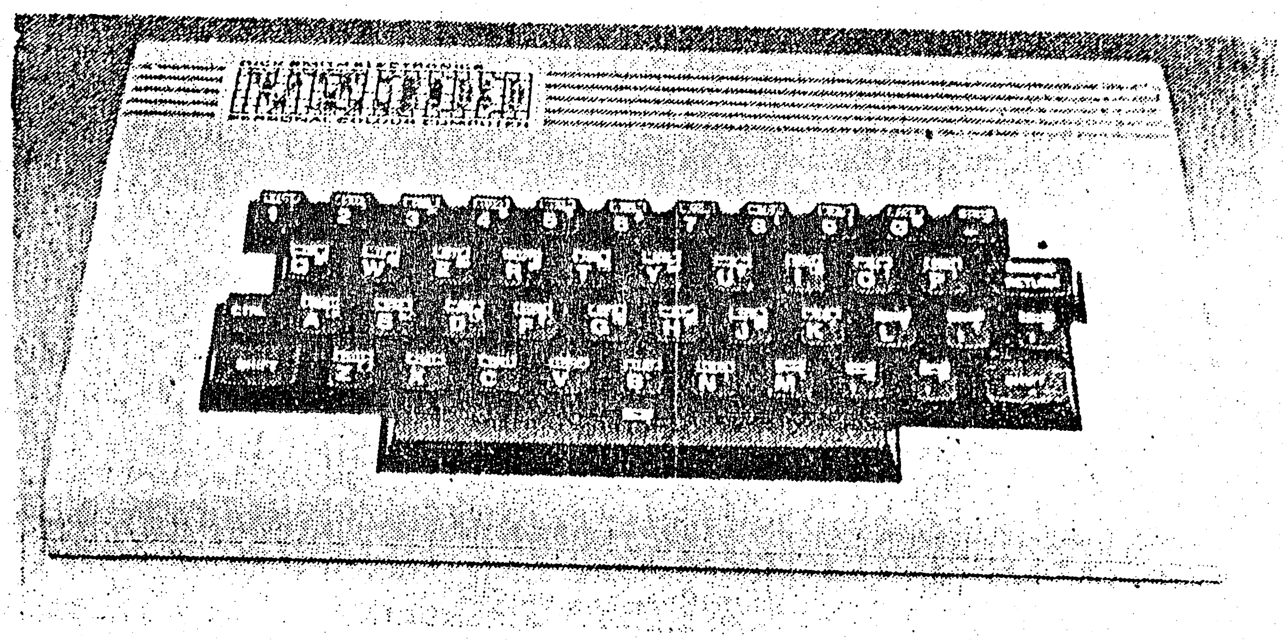
ISSUE NUMBER 4.

HUNTER VALLEY VZ USERS' GROUP NEWSLETTER.



SEPTEMBER 1986.

PRESIDENT..........Peter Ellis

VICE PRESIDENT.....Mark O'Brien

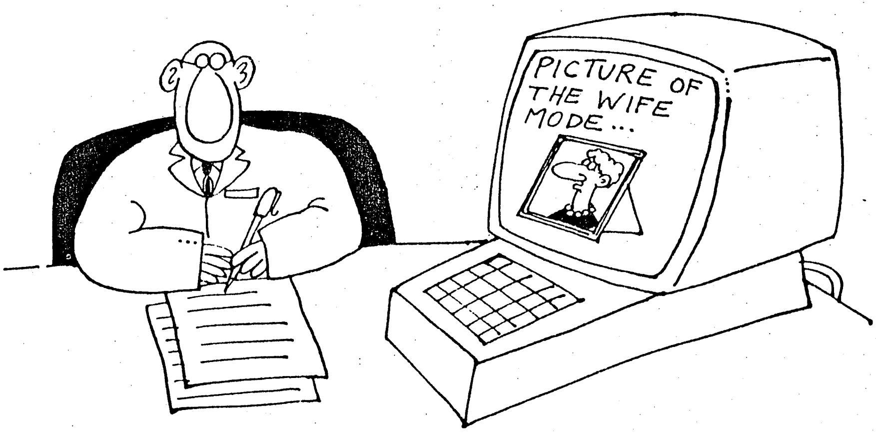
SECRETARY..........Ross Woods

TREASURER..........Leigh Rogers

CHAIRMAN/EDITOR....Gavin Williamson

Mail all submissions to:-

The Secretary, HVVZUG.,
Ross Woods,
83 Lambton pde.,
SWANSEA HEADS. 2281



Oh well, here we go again. I was'nt going to put anything in this month, but Gavin threatened me with greivious bodily harm if I did'nt. As you no doubt know last months meeting was a shambles (sorry about that chief). we had big printing problems with the newsletter as our regular printer is away at the moment and Gavin called on everybody except the Army to get it done. After a lot of hassles he managed to produce enough copies for the meeting only, so all you subscribers who failed to get your copy last month will recieve two months this time around and I thank you for being so understanding.

Having had quite a few Datasette problems over the years, mainly the last five months, and not being very impressed with the DR20 which I consider to be overpriced and unreliable I bought an Aquarius data recorder for use on the VZ300. I'm happy to say it's a Bloody Ripper. It loads everything I put into it regardless of which machine the program was recorded on, all this for $9.95. As you may know, what is recorded on a tricky dicky Datasette will not always play back on a non tricky dicky brand, but it seems the aquarius will accept almost anything. It would be ideal as a backup but you may be lucky to get one as they're as scarce as pork at a Bar Mitzvah.

That last paragraph was written on Thursday, on Friday I found out that there are still a limited number of those Datasettes still available but if you want one you had better be quick as they are going fast. Now, more news about the VZ300 program books that we have been waiting for with Bated Breath for the last twelve months. Wait for it, they have finally arrived (Gasp). After eagerly browsing through the two newies I came away disapointed. It seems that D.S.E are in such a deep rut they can't climb out of it. These New books are nothing more than the old VZ200 books recovered and are three years old at the very least. What's happened at Tricky Dickies. Can it be they're scared to bring anything new out in case Paul Keating Taxes it or is it just Apathy regarding the VZ. Am I crying in the wilderness or will I be heard down there in Wonderland. Stay tuned for the next boring episode. All jokes aside though, I will say that even old books are better than no books to someone who has nothing better to look to for help so 5 out of 10 for trying to D.S.E.

Have you noticed the price rises on the VZ gear lately. This one we'll certinaly blame on everybodies friend Mr.K. (the blames gotta go somewhere and he's handy). Ok,I've had my whinge so now you can do more important things,like reading the rest of this newsletter. See you in the funny pages.

Pete.

From small acorns, do mighty oaks grow. Well something like that, anyway. By this I don't mean that the founding members of the Hunter Valley VZ Users' Group are nuts, but, a recent update of our Membership list shows that we are now thirty strong. Great stuff! And to that we have Subscriber type Members, from the Central Coast, Rockhampton, Adelaide and country N.S.W.

Two contacts made in Queensland during the last month, the first, Dave Mitchell, who will be writing articles for our Newsletter on a regular basis (suprise Dave), and also initial contact has been established with Bob Kitch from Kenmore (near Brisbane I think). Bob has collected a massive amount of articles on software, hardware or anywhere to do with the VZ. It's also possible that Bob will become a contributor as well. Thanks for your letters Gentlemen. Who said the only good thing to come out Queensland was the main road?, they were wrong. It's VZ articles and the main road. Looking foward to a first article from both of these Members (soon).

The GROUP is currently considering purchasing a few cheap cassette recorders with the aim of sending one to each of the proven regular contributors. All articles could then be typed on the Word Processor program, the files saved to tape, and sent to me. The idea of every one using the same type of cassette player is to try and get away from the huge problem of loading errors. This is, in the main, a lot to with incompatability of cassette players.

Sorry folks, but there is no games review in this Issue. Our resident Games Reviewers are away on school holidays.

Seems the couple of worries that I wrote of in last months editorial have been solved. Thanks folks!

Great to see a couple of Mums along at the last meeting. Leigh, our (to date anyway) female Member, should have felt a little more at ease. (And a little less of Chris. HA!)

Seen or heard anything funny to do with computers lately? Are you a closet cartoon fan. We decided two issues ago to put some funnies in, but sadly, we are running out of them. Help!

Thanks to all who helped "save the day" with the assembling of the last Newsletter. Also thanks to Kevin Linsell, who showed up for the first time, last meeting, and ended up running off our spare and subscriber copies.

Inside you'll find the second section of the article by Robert Quinn, this will probably be a five part article. Also we have put another program in the Beginners Section. There are two interesting reviews on add-on gear for the VZ. The serial interface as reviewed by Ross Woods, and the Radio Teletype (RTTY) kit from Rudy, reviewed by Peter Ellis. Thanks fellas!

Just a final point to remember, thats if you, or if you hear of anyone who lives too far away, (or has other commitments) to attend our first thursday of every month meetings at St.Marks Church hall in Islington, then remember we now have the facilities to handle Subscribers. A few phone numbers listed below:

Gavin (049) 621678
Pete (049) 695697
Ross (049) 712843

BATTERY BACK-UP RAM BY DAVE BOYCE

This description will cover Battery Back-Up static RAM in any location, although this article is set up for (decoding only) in the 6000H to 67FFH area. That is the 2K gap left by the Disk Controller. Non Disk users can put an extra 8K RAM in the 4000H to 5FFFH area. Inside the VZ200 the 74LS139 has two unused decoded outputs, Pin 6 (8K Block) and Pin 12 (2K Block). Pin 12 is the one used in this article although an alternate decoder (74LS138) is shown for persons not wishing to open up their VZ200, or for VZ300 owners. For alternate 8K decoding see previous issue of this newsletter.

The operation of DATA retention is fairly easily achieved. Refering to the circuit diagram locate the 4 Way DIP Switch. You will find that only 3 Switches are used for the Battery Back-Up feature. S3 (Switch 3) is not used and no connection is made to it.

S1 - Controls the main power supply from the VZ to the RAM.
S2 - Controls the +3V (2 X AAA) Battery Back-Up Supply.
S3 - No connection to this Switch and is not used.
S4 - ENABLES or DISABLES the WRITE line from the VZ to the 2K RAM Chip.

Ensure that S1 is ON when the VZ is ON or You will not be able to WRITE to the RAM.

To WRITE to the RAM S1 and S4 must be ON.

To PROTECT RAM DATA while the VZ is ON, turn OFF S4.

To PROTECT RAM DATA while the VZ is OFF :- Before turning the VZ OFF ENSURE that S2 is ON and S4 are OFF.

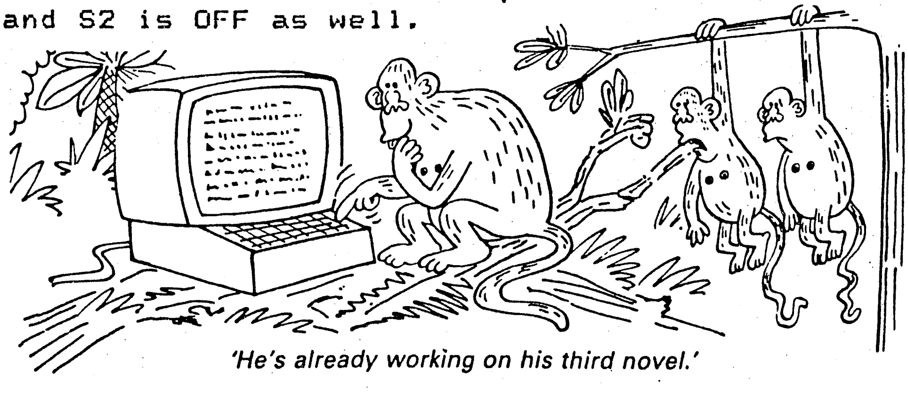
To ERASE DATA while the VZ is OFF, turn S2 OFF.

To ERASE DATA while the VZ is ON :- Turn S1 and S2 OFF. Wait approx. 10 sec's. before turning S1 back ON. S2 is optional.

NOTE -- Before REMOVING or REPLACING the Batteries ensure that S2 is in the OFF position.

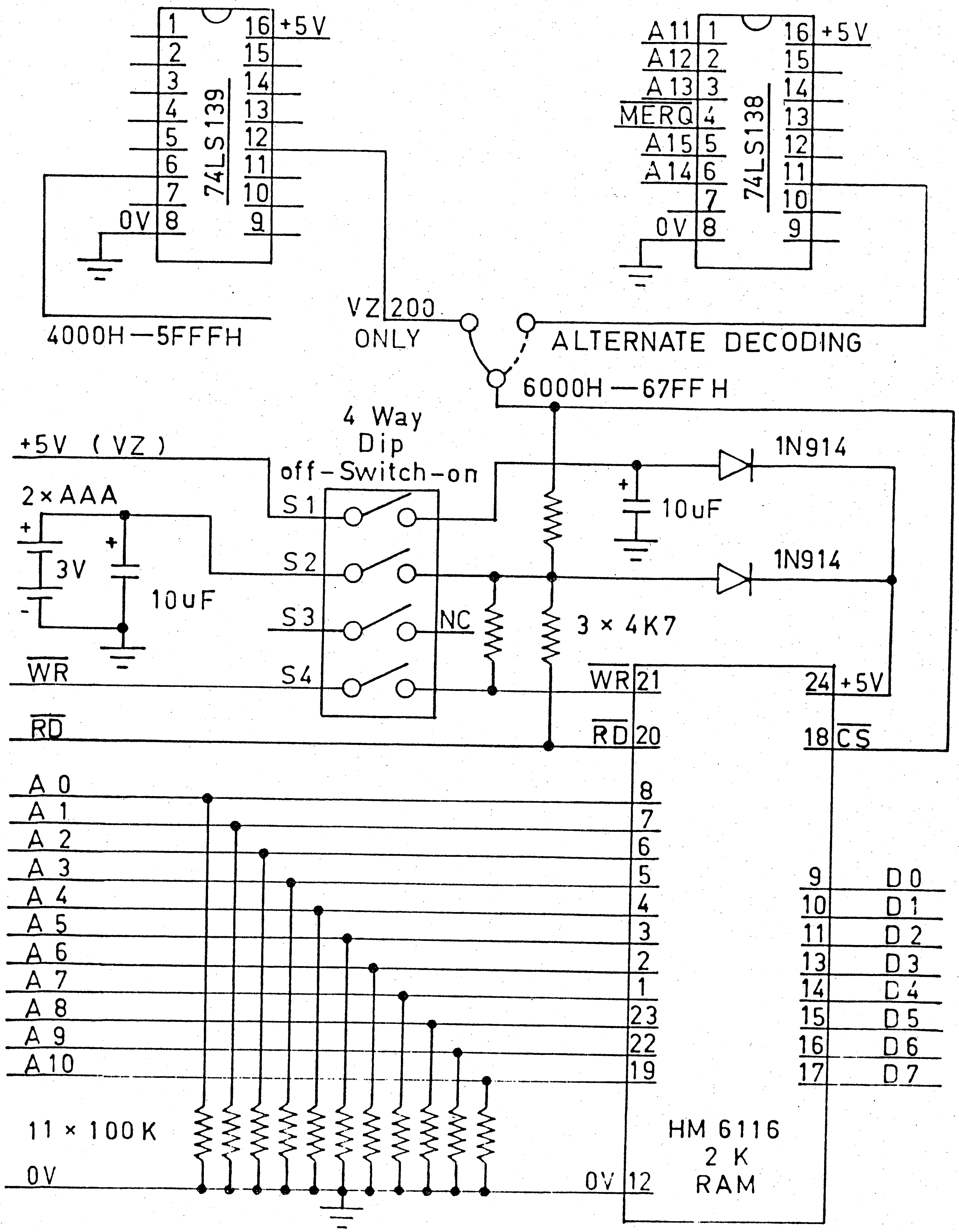
Use ALKALINE batteries only. (NOT Nickle Cadmium).

NOTE -- DO NOT REMOVE the 2K RAM Chip from it's Socket UNLESS the VZ is OFF and S2 is OFF as well.



'He's already working on his third novel.'

VZ 200 / 300 BATTERY BACK UP RAM



© COPYRIGHT DAVE BOYCE

TO WRITE WITH VZ ON

S1	ON	(+5V)
S2	X	(+3V)
S3	X	(NC)
S4	ON	($\overline{WR}$)

TO PROTECT WITH VZ ON OR BEFORE TURNING VZ OFF

S1	ON	(+5V)
S2	ON	(+3V)
S3	X	(NC)
S4	OFF	($\overline{WR}$)

TO ERASE WITH VZ ON

S1	OFF	(+5V)
S2	OFF	(+3V)
S3	X	(NC)
S4	X	($\overline{WR}$)

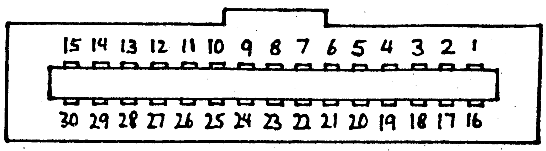
TO ERASE WITH VZ OFF

S1	X	(+5V)
S2	OFF	(+3V)
S3	X	(NC)
S4	X	($\overline{WR}$)

X = DONT CARE $\overline{WR}$ = WRITE NC = NO CONNECTION

CORRECTIONS — "*" SHOWS DATA INADVERTENTLY OMITTED

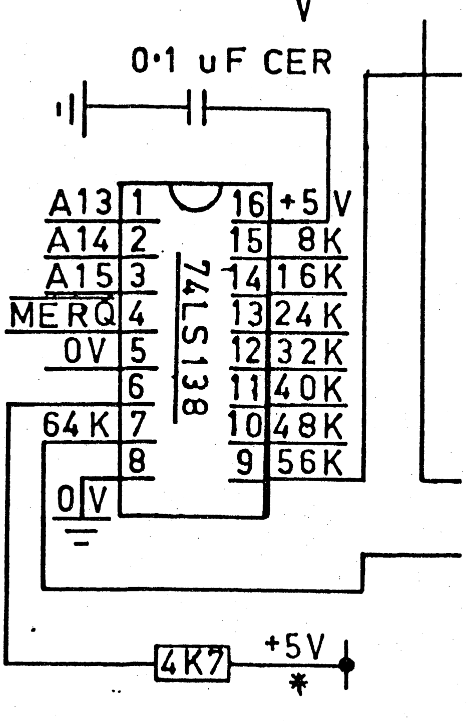
ISSUE 1 — PRINTER/ J'STICK PORT



PRINTER — JOYSTICK INTERFACES, ETC.

PIN NO. FUNCTION	PIN NO. FUNCTION
1 – N.C.	16 – N.C.
2 – N.C.	17 – N.C.
3 – N.C.	18 – N.C.
4 – +5VDC	19 – +5VDC
5 – $\overline{IORQ}$	20 – N.C.
* 6 – D3	21 – D4
* 7 – D6	22 – D5
* 8 – D2	23 – D7
* 9 – D0	24 – D1
* 10 – A5	25 – $\overline{RD}$
* 11 – A2	26 – A1
* 12 – A6	27 – A4
* 13 – A3	28 – A0
14 – $\overline{WR}$	29 – A7
15 – GND	30 – GND

ISSUE 3 — VZ300 MEMORY EXPANSION



SERIAL INTERFACE

The VZ Serial Interface is sold as an "add-on" kit project from D.S.E. It is priced in the catalogue at $49.95 and is CAT# K-6317.

When constucted, this kit, with the addition of a modem and telephone line, allows the VZ200/300 to communicate with electronic bulletin boards and data bases.

I constructed the kit and borrowed a modem, and soon managed to increase my telephone bill by ringing up databases out of the local phone area. The only database I contacted in Newcastle was the Micro-computer club. This database allows electronic mail, questions and answers, plus other information.

The Serial Interface turns the VZ into a "GLASS" terminal. It must be noted that this kit only allows terminal operation; ie: no saving to disk or tape and no transfer of memory contents. However, it does allow the VZ to control another computer, from anywhere in the world, via a telephone line and modem. The information can be sent from the VZ and recieved back on both screen and printer.

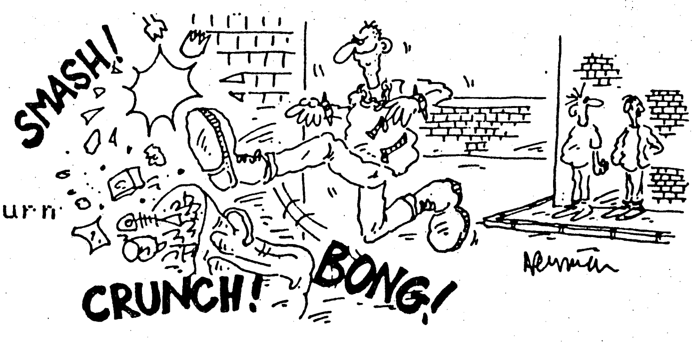
Physical details of the kit. The interface plugs into the memory expansion port of the VZ, the case supplied is identical to that of the memory expansion module. The connection to the modem is via a 25 pin RS232 plug fitted to the rear of the case. Only three connections are used, these are EARTH, RECIEVE and SEND. I also fitted a switch to the recieve and send lines to allow reversing the function. This was not part of the kit. This enables the interface to drive a serial printer directly (no modem) or, for that matter, any other serial devices directly.

I experienced no major problems in constucting the kit. I would suggest that some experience in fine/close soldering be obtained, before tackling this kit. It is rated by D.S.E. as (three star) ***, construction difficulty.

No terminal program is required. The program is in ROM and when the Serial Interface is connected, it is available on power up of the VZ. This unit runs at the speed of 300 BAUD.

MENU OPTIONS.

(0) enter terminal
(1) full - half duplex
(2) toggle printer: on/off
(3) set data bits: 8 or 7
(4) set stop bits
(5) set parity: odd, even or none
(6) add line feed to carriage return
SHIFT X to exit terminal.



'He could have been one of the great games writers of all time.'

Ross Woods.

PRINTER SHARER, COMPUTER SHARER & SERIAL BOARD.

This month I will (try to) describe in a little more detail the above three units from Don McKenzie. This is, of course, a continuation of last months article on the PBUFF.

PRINTER SHARER.

This unit will allow the user, at the flick of a switch, to access two printers from the output of the Buffer, or from the printer interface alone. This saves having to power down while you change the printers over. But, you must also remember that generally, while one printer is working (printing), you can not switch over. There is an exception to this rule though - see below in the PBUFF additions under HARDWARE PAUSE. These boards may be 'stacked' which will allow more printers (or anything that looks like a printer e.g. speech) to fed from the one Buffer.

My own system consists of : VZ300 (RAM to FFFFH), Datacassette, Disk Drive, Joysticks, 64K RAM Printer Buffer, Printer Sharer, Dot Matrix Printer and PP-40 Printer Plotter.

COMPUTER SHARER.

This unit is in effect a Printer Sharer in reverse. This allows two computers to feed into one printer, or looking further, feed into a PBUFF which will run a printer or into a Printer Sharer. The computers need not be the same e.g. a VZ and a C-64 (with interface). These boards may also be 'stacked', this will allow more computers to feed into a PBUFF or Printer Sharer.

SERIAL BOARD. RS-232.

This unit will allow the user who has either :-

(1) Serial Input only Printer
(2) Serial out only Computer
(3) or Serial only System

to use the PBUFF and its add-ons with practically any of RS-232 / Parrallel (Centronics) computer ann printer combination.

PBUFF ADDITIONS. Software and Hardware Pauses.

The software pause is activated by sending a double backslash \\. Sending this string will stop the PBUFF in its tracks. You can deactivate this command if necessary, e.g. you may have to send this string to a printer (or some other device).

The hardware pause can be used to stop the printer at any time. Only while the hardware pause is activated may you change over to the other printer.

The printer sharer and computer sharer boards with instructions cost $12 each and approx. $54 to construct. The RS-232 serial board with instuction is $18 to buy, construction $-$?

For more information on these three units and the PBUFF, contact:-

Don McKenzie,
29 Ellesmere Cres.,
Tullamarine 3043 'phone 03 3386286

ARTICLE WRITTEN by DAVE BOYCE.

UNDERSTANDING YOUR VZ . . . PART TWO.

THIS IS A CONTINUATION OF THE ARTICLE STARTED IN THE AUGUST EDITION.

An address pointer stores the address it 'points' to in a LO, HI format according to the formula:

address = LO + HI x 256

where LO is the number (the LO byte) stored in the first of the memory cells making up the pointer and HI is the number (the HI byte) stored in the second cell. To determine the address pointer we need to know the addresses of the two cells of the pointer. Then we can PEEK at those cells to find LO and HI. 30884 30885 are the addresses of the START OF PROGRAM pointer. So, if you switch on your VZ and enter this command:

```
PRINT PEEK(30884) + PEEK(30885) x 256
```

you will discover the start of program address.

You got 31465? That is the usual start address of program memory, which follows on from the end of the communications region. When we enter basic program lines, whether we type them in using the keyboard, CLOAD them from tape or LOAD them from disk, the lines take up residence in RAM from this point on.

31465 is the start address for the line with the lowest line number in a BASIC program. Lines are stored in the sequence of their line numbers. Add a new line, and all the lines with higher line numbers will be pushed up in program memory to create a slot for the new line. Delete a line, and all the higher numbered lines will be pulled down to close the gap.

To explore the structure of program lines in memory, enter this two line program and RUN it:

```
10 FOR A=31465 TO A+37:PRINT PEEK(A);:NEXT

20 PRINT:LIST-20
```

The program has listed itself in two different ways. Line 20 performed the familiar BASIC listing of the program. Line 10 produced a PEEK listing of the program as a sequence of numbers stored in memory cells from 31465 onwards. How do the two compare? In the BASIC listing each program line has a number, a set of BASIC words (FOR, =, TO, PRINT, PEEK, etc.) and various other characters (digits, colons, brackets, A, etc.) which, together with the associated BASIC words, make up BASIC statements.

Now add these lines to the program:

```
30 CLS:FOR A=31465 TO A+37:X=PEEK(A)

40 IF X>31 AND X<95 THEN PRINT CHR$(X);ELSE PRINT X;

50 NEXT:PRINT:PRINT:GOTO 10
```

and RUN 30.

UNDERSTANDING YOUR VZ . . .PART TWO.

The top PEEK listing provides an ASCII decoding of those bytes that fall within the ASCII range. Variables, digits, brackets, colons, semicolons come through, and if we compare the two PEEK listings it is not hard to figure out where the line numbers occur. But where are the BASIC words? Their positions in the lines seem to be occupied by triple-digit numbers. Apparently more decoding is necessary. Add this line to the program and RUN 30 again:

5 REM A TEST OF THE ASCII PEEK LISTING

The text of the REM statement gets thorough ASCII decoding, letter by letter, blank by blank. Even the BASIC word PEEK is ASCII decoded, though it was not decoded in the previous RUN.

The text of a BASIC line always begins with the fifth byte for that line. The third and fourth bytes are the line number. Only two bytes (no more, no less) are used to record how many digits the line number, using the LO HI format, no matter how many digits the number may have:

line number = byte 3 + byte 4 x 256

Because the line numbers in this program are small, byte 4 is '0', which is 0 x 256. The first two bytes of a BASIC line record the start address of the next line in the program, again according to the LO HI formula:

start address of line = byte 1 + byte 2 x 256

If we delete line 5 and RUN 30 we see that the first two bytes of line 10 are '4' and '123' which gives us the start address for line 20 as 4 + 123 x 256 = 31492.

PRINT PEEK(31492) gives us 15. If we locate '20' in the PEEK listing as the third byte of line 20, we see that byte 1 of line 20 (two locations back from the number 20) is '15'. But if '15' is the first byte of line 20 then the '0' to the left of '15' must be the last byte of line 10. Indeed, the last byte of every line is a '0', a marker indicating that this line ends here. There is even a pseudo-end of line marker in the memory cell preceding the first byte of the first line of a program -- a zero at address 31464, the last cell of the communications region. If this byte is POKE changed to a non-zero value, your program will not RUN.

Note that in the REM statement PEEK occupied four bytes of memory, one byte for each character:

```
80   69   69   75
P    E    E    K
```

In line 10, however, the position of the PEEK word is occupied by a triple-digit number -- 229.

UNDERSTANDING YOUR VZ . . . PART TWO.

This byte '229' is a token code for PEEK. Every BASIC word is represented in text of program lines (once they have passed into program memory) by a unique token code, a number between 128 and 250. A one byte token uses up three memory cells less than the four character word PEEK. And since BASIC words constitute a major part of most programs, tokenisation results in considerable saving of memory occupied by a program. In the text of a REM, PRINT or LPRINT statement, words like PEEK, FOR, GOSUB are not regarded as BASIC words; they are just strings of characters to be displayed and are not tokenised. Inverse and graphic characters are also coded using numbers greater than 127. They are distinguished from the token codes for BASIC words by their relation to quotes (""). Within quotes such numbers are interpreted as character codes; outside quotes they are token codes.

One peculiarity is the ELSE statement. ELSE is always preceded in memory by a colon (:), but the colon is never displayed when lines are LISTed or LLISTed. You have to count this invisible colon to get an accurate estimate of the length of a line involving ELSE statements.

Enter the TOKEN routine and RUN. The routine will display all the basic words that make up the VZ's BASIC VOCABULARY, along with their token codes. Press any character key to halt the display. Press again to continue. If you have a PP40 (or TP40) printer then Token will LPRINT a hard copy of all the words and their codes if you RUN 100.

T O K E N.

```
10 CLS:D=127:SOUND 0,1:FOR R=5712 TO 6175:B=PEEK(R)
20 A$=INKEY$:A$=INKEY$:IFA$<>""THEN C=NOT C:SOUND 30,1
30 IF C=-1 THEN 20
40 IF B>128 THEN D=D+1:PRINT:PRINT D;
50 IF B>169 THEN PRINT CHR$(B-128);ELSE IF B=0 THEN 80
60 IF B>31 AND B< 96 THEN PRINT CHR$(B);
70 SOUND 0,1
80 NEXT
100 CLS:LPRINT CHR$(18)"S1,"CHR$(17):D=127:FOR R=5712 TO 6175
110 B=PEEK(R):IF B>169THEN D=D+1:PRINT:PRINTD,CHR$(B-128);:C=C+1
120 IF C=1 AND B>169 THEN LPRINT ELSE IF C>1 THEN LPRINT,:C=0
130 IF B>169 THEN LPRINT D;CHR$(B-128);ELSE IF B=129 THEN PRINT
135 PRINT D;:D=D+1
140 IF B>31 AND B<96 THEN PRINT CHR$(B);:LPRINT CHR$(B);
150 NEXT
```

PART 3 NEXT MONTH...

```
100 CLS:COLOR 3:PRINT:PRINT:PRINT
110 PRINT"████████████████████████████"
120 PRINT
130 PRINT"     MATHS TEACHER"
140 PRINT"     ========================"
150 PRINT"     ADAPTED TO VZ BY F.ELLIS"
160 PRINT"     FOR HUNTER VALLEY VZ USERS      GROUP":PRINT
170 PRINT"████████████████████████████"
180 FOR Z=1 TO 2000:NEXT
190 INPUT"WHAT IS YOUR NAME"; N$
200 PRINT" OK ";N$;", HOW MANY QUESTIONS":INPUT A
210 PRINT" WHAT LEVEL WOULD YOU LIKE ";N$
220 PRINT" (1) BEGINNER":PRINT" (2) HARD":PRINT" (3) FRUSTATING"
230 INPUT"ENTER 1,2 OR 3";L
240 CLS:PRINT:PRINT:PRINT
250 PRINT"ENTER (1) ADDITION":PRINT"       (2) SUBTRACTION"
260 PRINT"       (3) DIVISION":INPUT"       (4) MULTIPLICATION";B
270 PRINT"USE OF CALCULATORS IS CHEATING":FOR Z=1 TO 2000:NEXT
280 IF B=1 THEN B$="ADDITION":GOTO 330
290 IF B=2 THEN B$="SUBTRACTION":GOTO 480
300 IF B=3 THEN B$="DIVISION":GOTO 570
310 IF B=4 THEN B$="MULTIPLICATION":GOTO 720
320 PRINT"ILLEGAL CHOICE - WE'LL START    WITH ADDITION."
330 FOR N=1 TO A
340 PRINT"QUESTION #";N
350 E=RND(1000):F=RND(1000)
360 IF L=1 AND (E>100 OR F>100) THEN 350
370 IF L=2 AND (E>500 OR F>500) THEN 350
380 PRINT E;" + ";F;" = ";
390 INPUT G:H=E+F
400 IF G=H THEN 440
410 PRINT"NO, THAT'S NOT CORRECT"
420 PRINT"THE ANSWER IS";H
430 I=I+1:GOTO 460
440 PRINT"ABSOLUTELY CORRECT"
450 C=C+1
460 NEXT N
470 GOTO 810
480 FOR N=1 TO A
490 PRINT"QUESTION #";N
500 E=RND(1000):F=RND(1000)
510 IF L=1 AND (E>100 OR F>100) THEN 500
520 IF L=2 AND (E>500 OR F>500) THEN 500
530 PRINT E;" - ";F;" =";
540 INPUT G:H=E-F
550 IF G=H THEN 440
560 GOTO 410
570 FOR N=1 TO A
580 PRINT"QUESTION #";N
590 E=RND(12)
600 IF L>2 THEN E=E+RND(E)
610 IF E<1 THEN 590
620 IF L=1 THEN F=RND(12)*E
630 IF L=2 THEN F=RND(100)+(RND(12))
640 IF L=3 THEN F=RND(100)*(RND(12))*E
650 H=F/E
660 IF L=2 AND INT(H) <> H THEN 630:GOTO 690
670 IF L=3 AND INT(H) <> H THEN 640
680 PRINT F;" / ";E;" = ";
690 INPUT G
```

```
700 IF G=H THEN 440
710 GOTO 410
720 FOR N=1 TO A
730 PRINT"QUESTION #";N
740 IF L=1 THEN E=RND(12):F=RND(12)
750 IF L=2 THEN E=RND(12):F=RND(12)+RND(12)
760 IF L=3 THEN E=RND(12)+RND(12):F=RND(12)*RND(12)+RND(12)
770 PRINT E;" X ";F;" = ";
780 INPUT G:H=E*F
790 IF G=H THEN 440
800 GOTO 410
810 FOR Z=1 TO 1000:NEXT:CLS
820 PRINT"     F I N A L  S C O R E"
830 PRINT"     --------------------":PRINT:PRINT
840 PRINT"CORRECT ANSWERS   ";C
850 PRINT"INCORRECT ANSWERS ";I
860 PRINT
870 T=C/A*100:PRINT N$;" FOR THIS EXERSISE YOU"
880 PRINT"ATTEMPTED";A;"QUESTIONS AT LEVEL";L;"IN ";B$;"."
890 PRINT"YOU ACHIEVED ";T;"%"
900 END
```

Well here we go at trying to give you a bit of an explanation about the preceeding program in this months BEGINNERS SECTION. I'll only go over those lines/routines where I feel an explaination is required. There are in fact many programs like this around, but this is a very simple version. It can be lots of fun, when you have mastered it, test the rest of the family.

110 SHIFT + J 32 times.
130 4 SPACES then in INVERSE alternated with a space
180 a delay loop
260 CAREFUL - the last one of these is an INPUT
320 if you press any other key than those you are told to, then the game will go straight into ADDITION
350 makes sure the questions are totally RANDOM
360 & 370 sets levels, that is relation to what you pressed at the start
410 to 460 this routine is used every time you answer a question, to see if you get it right or wrong
450 keeps track of all the questions that you answer correctly
480 goes to the "GAME OVER" routine when all questions have been attempted
490 to 560 as above, but for subtraction
570 to 710 again, as above, but for division
610 makes sure you don't get asked division by ZERO questions
660 & 670 checks that all questions asked will have WHOLE NUMBERS (intergers) as answers. Don't want things like 158/157, do we?
720 to 800 routine for multiplication
820 to 890 game over routine. Tallies up how many answers wrong and right, then works this out into a percentage form

CHEAPER THAN PHONES AND NOT AS MESSY AS PIGEONS!

Some time ago D.S.E introduced the VZ R.T.T.Y Kit for people who are into Radio as a hobby. This Kit proved to be a bit of a flop because it had only one baud speed. Now as any Ham knows, one Baud speed is as useful as a toothache. A good mate of mine, also a club member and a local Ham radio operator well known in the Newcastle area (All this in one bloke) is Rudy Meinsma VK2FIM. Rudy decided to go a couple of steps further and developed his own unit. This unit has been up and running for some time now and is proving quite popular in the Hunter Region.

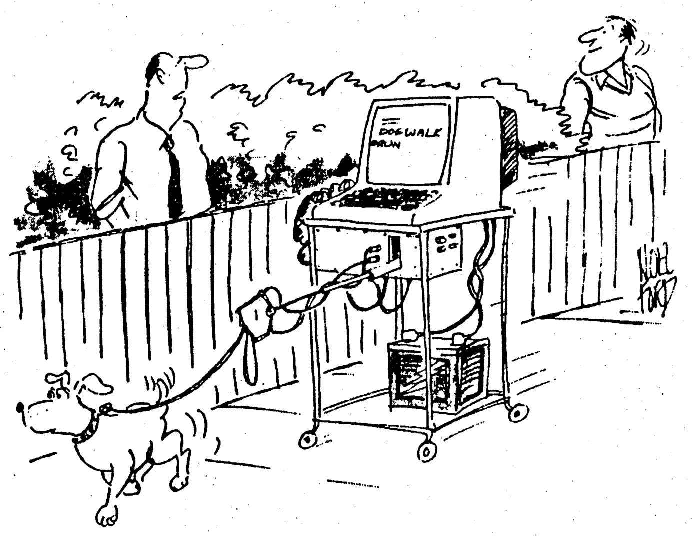
This VZ200-300 rtty interface also handles morse code. The P.C.B, outlay of the P.C.B, circuit, partslist, and instructions are now available to enable you to construct a superior unit which will give exellent results.

The main features are:

1..split screen display for TX and RX
2..fully buffered trarsmit keyboard with 1000 char. buffer.
3..nine message memories which can be saved perm. on tape or disk with the software prog.
4..speed is key selectable from 45 Baud to 99 Baud for rtty and 5 to 99 wpm for morse.
5..ability to type in TX buffer while receiving.
6..line printer driver.
7..sync idle on transmit.
8..morse sent as audio to both internal speaker and cassette output.
9..shift can be swicthed from 170HZ-425HZ to 850HZ to tune in on commercial rtty or news broadcasts.

The circuitry is quite simple and should be no problem in construction. 3 ICs, 5 transistors, 3 preset pots, LED for fine tune and a handful of resistors and capacitors.

Cost of the P.C.B and necessary paperwork is $18 plus $1 post. If you are interested you may phone Rudy on (049) 437548.



'You actually make use of it, then — not just for playing games.'